

```
import java.sql.*;
import java.util.Random;
```

```
public class
DatabaseMultiThreadingProject {
```

```
// مسار قاعدة البيانات - سيتم إنشاء ملف باسم
university.db في مجلد المشروع
```

```
private static final String DB_URL =
"jdbc:sqlite:university.db";
```

```
public static void main(String[] args) {
// 1. تهيئة قاعدة البيانات وإنشاء الجدول
initializeDatabase();
```

```
System.out.println("=== بدء عمل المسالك
(Threads) بشكل متوازي ===");
```

```
// 2. إنشاء مسالك للإدخال (Insert Threads)
```

```
Thread insertThread1 = new
Thread(new InsertTask(), "Thread-
Insert-1");
```

```
Thread insertThread2 = new  
Thread(new InsertTask(), "Thread-  
Insert-2");
```

// 3. إنشاء مسالك للقراءة (Read Threads)

```
Thread readThread1 = new  
Thread(new ReadTask(), "Thread-Read-1");  
Thread readThread2 = new  
Thread(new ReadTask(), "Thread-Read-2");
```

// 4. تشغيل جميع المسالك في وقت واحد

```
insertThread1.start();  
insertThread2.start();  
readThread1.start();  
readThread2.start();  
}
```

```
private static void initializeDatabase() {  
    try (Connection conn =  
DriverManager.getConnection(DB_URL);  
        Statement stmt =  
conn.createStatement()) {  
        // إنشاء جدول الطلاب
```

```
String sql = "CREATE TABLE IF NOT  
EXISTS students (" +  
            "id INTEGER PRIMARY KEY  
AUTOINCREMENT," +  
            "name TEXT NOT NULL," +  
            "grade INTEGER)";
```

```
stmt.execute(sql);  
// تنظيف الجدول ليكون فارغاً عند كل تشغيل  
جديد (اختياري)
```

```
stmt.execute("DELETE FROM  
students");
```

```
System.out.println("تم إنشاء قاعدة  
البيانات وجدول الطلاب بنجاح");
```

```
System.out.println("-----  
-----");
```

```
    } catch (SQLException e) {  
        System.err.println("خطأ في تهيئة  
القاعدة: " + e.getMessage());  
    }  
}
```

// كلاس مهمة الإدخال (Insert Operation)

```
static class InsertTask implements
Runnable {
    @Override
    public void run() {
        String[] names = {"Ahmed", "Sara",
"Layan", "Zaid", "Noor"};
        Random random = new Random();

        for (int i = 1; i <= 3; i++) {
            try (Connection conn =
DriverManager.getConnection(DB_URL)) {
                String name =
names[random.nextInt(names.length)];
                int grade = 70 +
random.nextInt(30);

                String sql = "INSERT INTO
students(name, grade) VALUES(?, ?)";
                PreparedStatement pstmt =
conn.prepareStatement(sql);
                pstmt.setString(1, name);
                pstmt.setInt(2, grade);
                pstmt.executeUpdate();
            }
        }
    }
}
```

```
System.out.println("[إدخال بيانات]
```

```
 بواسطة " +
```

```
Thread.currentThread().getName() + " |
```

```
أضاف الطالب : " + name);
```

```
// تأخير بسيط لإظهار التوازي
```

```
Thread.sleep(600);
```

```
} catch (Exception e) {
```

```
System.err.println(" خطأ في
```

```
الإدخال: " + e.getMessage());
```

```
}
```

```
}
```

```
}
```

```
}
```

```
// كلاس مهمة القراءة (Read Operation)
```

```
static class ReadTask implements
```

```
Runnable {
```

```
@Override
```

```
public void run() {
```

```
for (int i = 1; i <= 3; i++) {
```

```
try (Connection conn =
```

```

DriverManager.getConnection(DB_URL)) {
    String sql = "SELECT COUNT(*)
AS total FROM students";
    Statement stmt =
conn.createStatement();
    ResultSet rs =
stmt.executeQuery(sql);

    if (rs.next()) {
        int total = rs.getInt("total");
        System.out.println("[ قراءة
بيانات بواسطة ]بيانات
Thread.currentThread().getName() + " |
العدد الحالي للطلاب
: " + total);
    }

    // تأخير بسيط لإظهار التوازي
    Thread.sleep(800);
} catch (Exception e) {
    System.err.println(" خطأ في
القراءة: " + e.getMessage());
}
}

```

```
}  
}  
}
```

هادا الكود غير يلي قبل؟ هادا مع تعديل
كيف تشغل الكود؟

قم بتحميل ملف SQLite JDBC Driver (ملف jar)
من الإنترنت.

أضفه إلى مشروعك (في Eclipse: رايت كليك على
المشروع -> Add External Build Path -> Archives).

قم بتشغيل الكود وستظهر لك المخرجات متداخلة، مما
يثبت التوازي.